

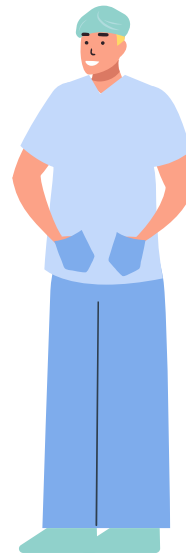
# FL Koala Product Presentation

## Clients:

- Radiation oncology department from Austin

## Team members:

- Lin Duan 1351342 – Product Owner
- Zhihao Liang 1367102 – Quality Assurance
- Lingyi Kong 1397061 – Scrum Master
- Yijun Liu 1132416 – Development Leader
- Junhao Kong 1133093 – Deployment Leader
- Yuncong Ji 1373110 – Architecture Leader



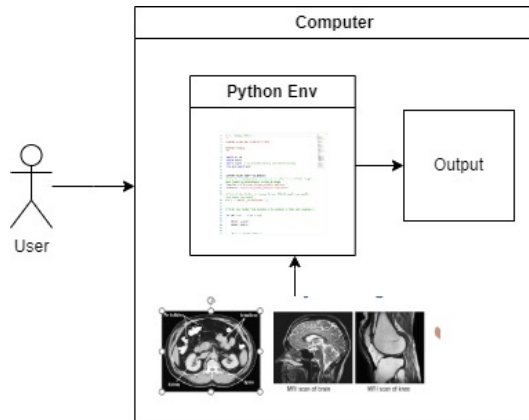
# Table of Content

- |   |                    |   |                    |
|---|--------------------|---|--------------------|
| 1 | Project Background | 4 | Project Management |
| 2 | Product Showcase   | 5 | Project Reflection |
| 3 | Current Progress   | 6 | Handover           |

# Project Background

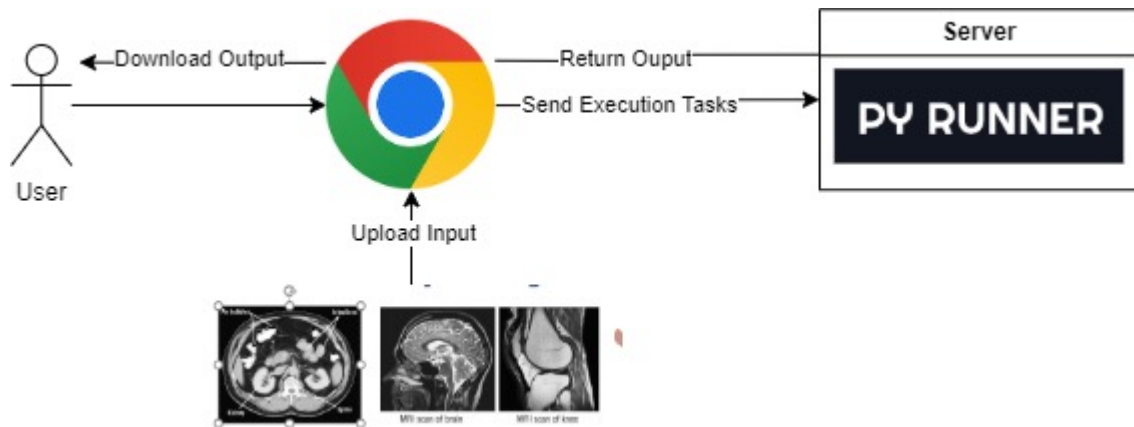
Struggling with installing  
Python Environments?

Have no idea about Python  
script?



- Radiation oncology department at Austin Health are using Python scripts to handle medical imaging such as CT and MRI
- **However, they are struggling to help their team member who doesn't know python to utilize those python analysis tool**

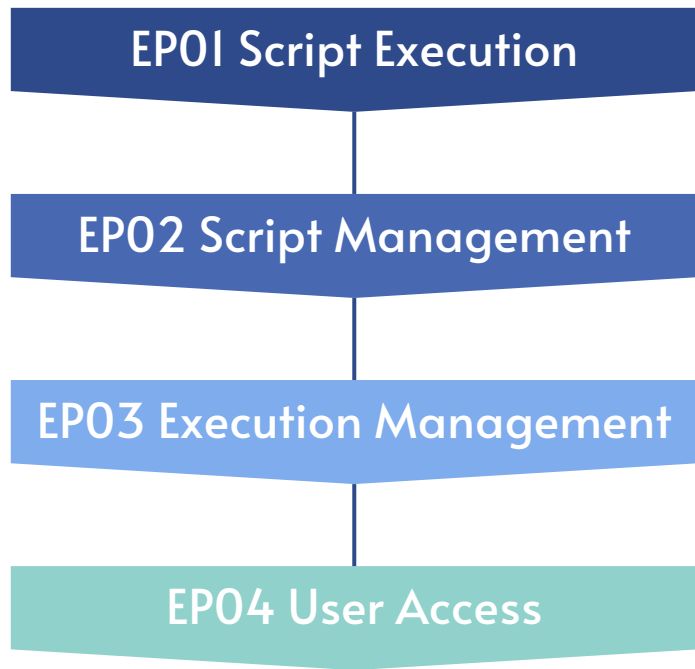
# Project Background



- Researchers now can easily use our product Pyrunner to run a python script with one click (or may be two clicks)



# What Our Website Can Do



## Run Python Script

Enable non-Python users to run Python scripts effortlessly at anywhere

## Manage Shared Scripts

Provide users their own script libraries and easy admin management.

## Check Execution History

Enable users to check execution history and outputs effortlessly.

## Manage Your Team

Allow flexible team management

# Product Presentation in Scenarios

## Scenarios I



**As**

Non python user

**I want**

Run specific python script without local installations configurations or Python knowledge

**So that**

I can perform CT/MRI image analysis tasks using python from any location easily

# Product Presentation in Scenarios

## Scenarios 2



**As**

Admin user

**I want**

Upload, modify, hide or delete scripts and manage scripts list easily

**So that**

Coworkers with proficient coding knowledge can provide needed scripts easily and the other group can focus on utilizing scripts

# Product Presentation in Scenarios

## Scenarios 3



**As**

Admin user

**I want**

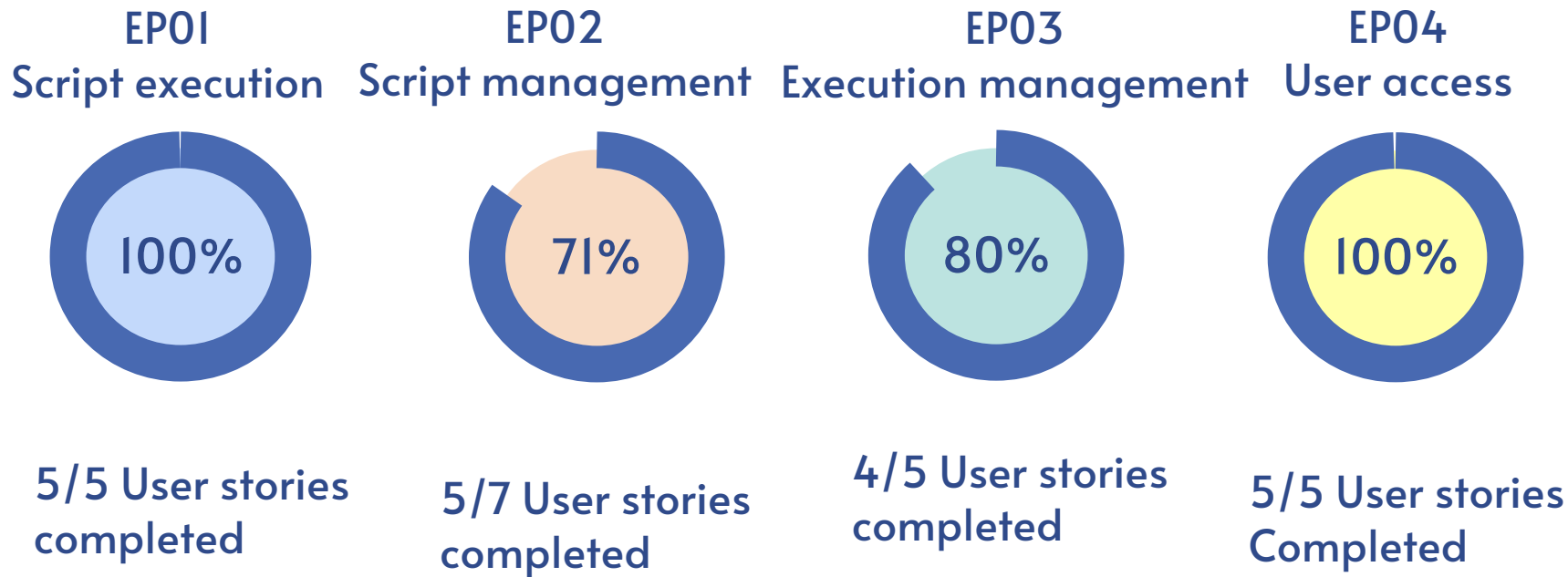
Easily setup different users' access based on their role and also could setup new account or reset for existing accounts

**So that**

The efficient management of users can be handles within teams.



# Current Status of Our Project



# Dynamic Requirements

The screenshot shows a web browser window with the URL `localhost:3000/dashboard/guide`. The page title is "User Guide for Python Code Running Platform". The left sidebar contains a "PY RUNNER" header and a navigation menu with items: Overview, Script List, Execution List, Account, Script Guide (highlighted), and an "Administrator Area" section with "User Management". The main content area has a heading "User Guide for Python Code Running Platform" and an "Overview" section. The Overview text states: "This platform allows users to execute Python code by uploading a zip file containing their input data. Each execution environment is set up with a pre-defined structure consisting of your main script ('main.py') and two directories ('input' and 'output') to manage data and results." Below this is a "File Structure" section with the text: "Upon execution, the user's environment will be structured as follows:" and a bulleted list: 

- `main.py`: The main Python script where your code will be executed.
- `input/`: A directory where the contents of your uploaded zip file will be extracted.
- `output/`: A directory where you can store output files generated by your code.

 A file explorer view shows a tree structure: 

- 137
  - input
  - output
  - main.py
  - output.zip
  - script.log

 Below the file explorer is the "Accessing Input Files" section. It includes "Step 1: Import the os Module" with the instruction "Start by importing the os module at the beginning of your main.py script:" and a code block: 

```
import os
```

 It also includes "Step 2: Get the Current Working Directory" with the instruction "Use `os.getcwd()` to get the path to the current working directory where `main.py`, `input`, and `output` are located:" and a code block: 

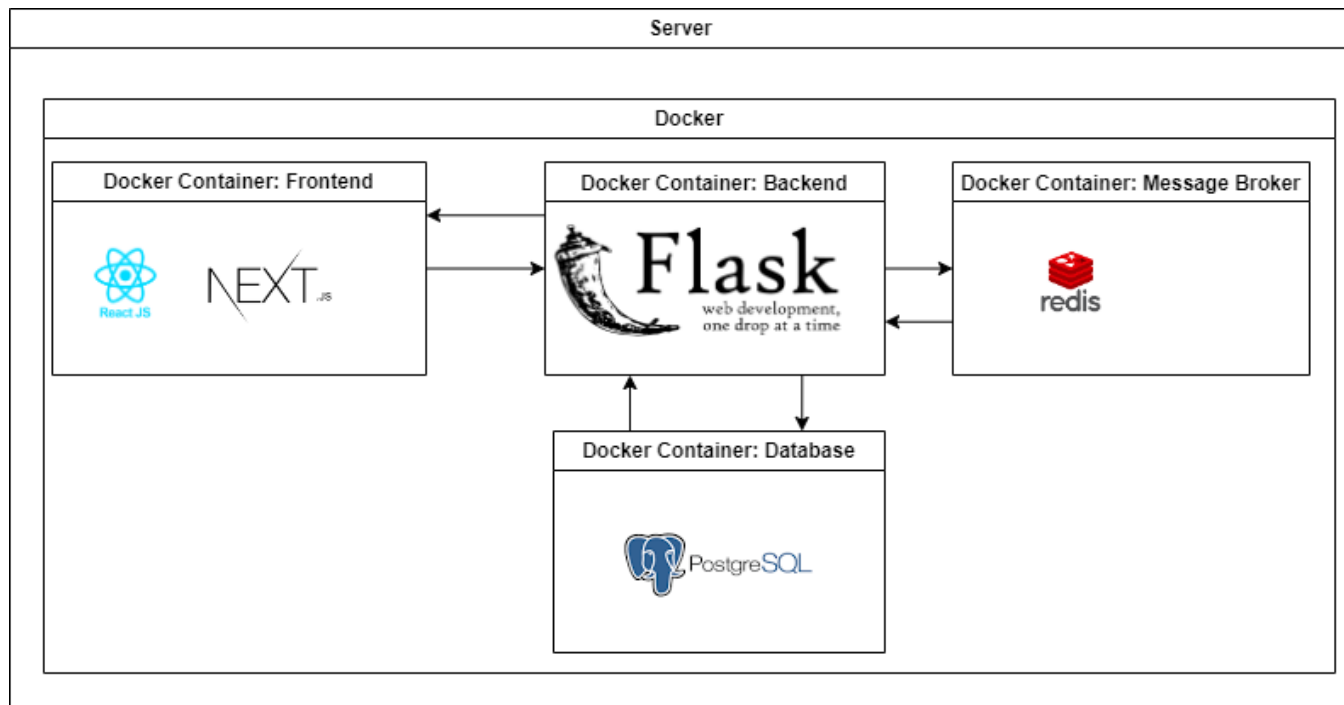
```
current_dir = os.getcwd()
```

 Finally, it includes "Step 3: Construct Path to the Input Folder" with the instruction "Append `/input` to the `current_dir` to form the path to the input directory:".

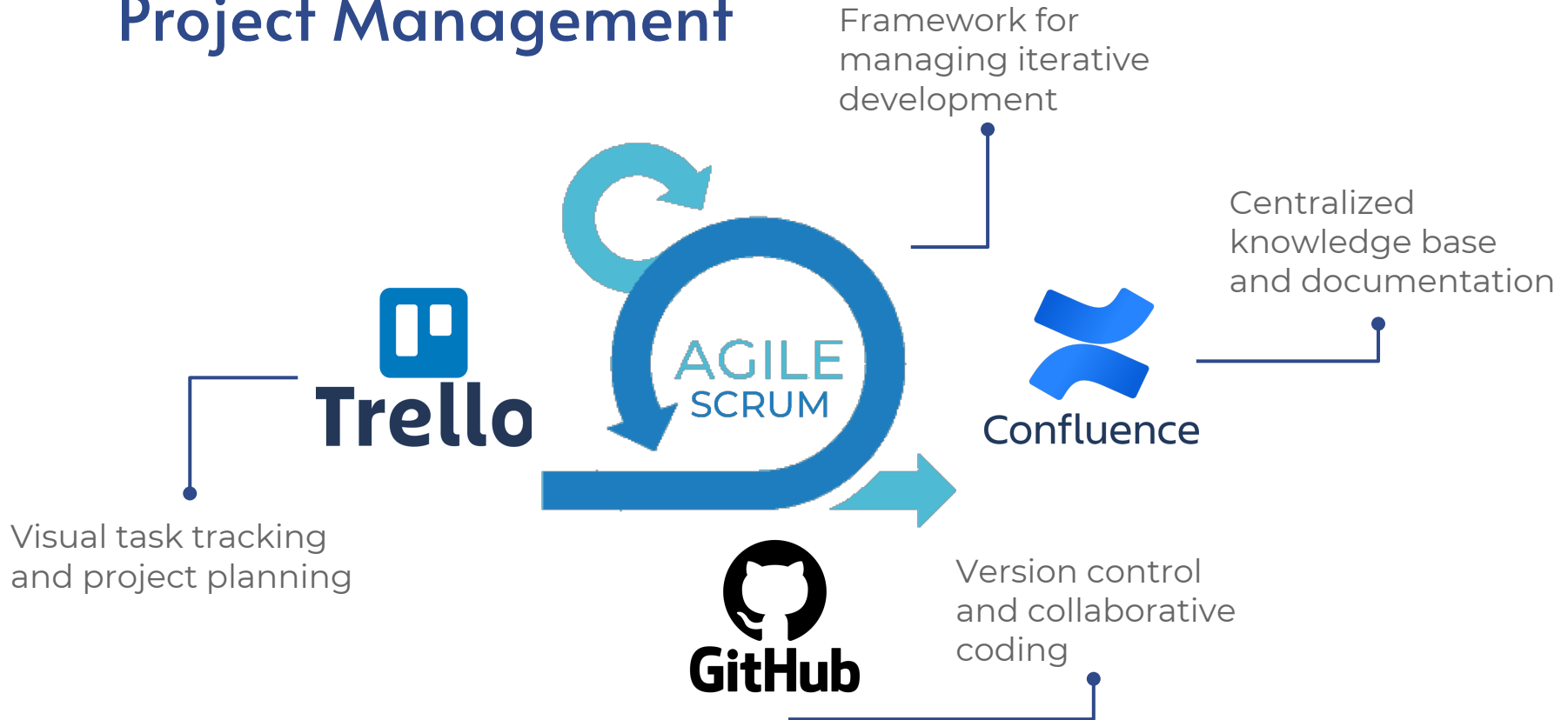
2/2 Requirements in Script management

2/2 New Requirement in User Access

# Product Architecture



# Project Management



# Project Reflection

PYRUNNER

Learn

- A. React
- B. Flask
- C. Agile
- D. Documentation
- E. Client  
Commucation

Challenge

- A. Input file size is  
massive
- B. Implement  
asynchronous  
tasks
- C. Team  
communication

Further Improvement

- A. Handle various  
execution input
- B. Duplicate a script

# Handover

1

Codebase, Access Rights and Deployment Instruction (GitHub)

2

Developer Documentation (Confluence and Trello)

3

User Manuals: Guides on how to use the application with step-by-step instructions.

# Q&A



**Thank You !**